



별첨 사본은 아래 출원의 원본과 동일함을 증명함.

This is to certify that the following application annexed hereto  
is a true copy from the records of the Korean Intellectual  
Property Office.

출원 번호 : 10-2003-0086048  
Application Number

출원 년 월 일 : 2003년 11월 29일  
Date of Application NOV 29, 2003

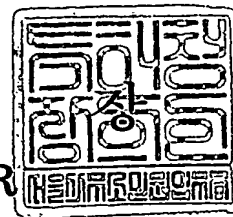
출원인 : 삼성전자주식회사  
Applicant(s) SAMSUNG ELECTRONICS CO., LTD.



2004 년 04 월 02 일

특 허 청

COMMISSIONER



BEST AVAILABLE COPY

## 【서지사항】

【서류명】	특허출원서
【권리구분】	특허
【수신처】	특허청장
【참조번호】	0020
【제출일자】	2003.11.29
【국제특허분류】	H04L
【발명의 명칭】	저밀도 패리티 검사 부호화를 위한 인터리빙 방법
【발명의 영문명칭】	Method of interleaving for low density parity check encoding
【출원인】	
【명칭】	삼성전자 주식회사
【출원인코드】	1-1998-104271-3
【대리인】	
【성명】	이영필
【대리인코드】	9-1998-000334-6
【포괄위임등록번호】	2003-003435-0
【대리인】	
【성명】	이해영
【대리인코드】	9-1999-000227-4
【포괄위임등록번호】	2003-003436-7
【발명자】	
【성명의 국문표기】	김기현
【성명의 영문표기】	KIM, Ki Hyun
【주민등록번호】	691220-1053119
【우편번호】	463-703
【주소】	경기도 성남시 분당구 구미동(무지개마을) 대림아파트 103동 1103호
【국적】	KR
【발명자】	
【성명의 국문표기】	이윤우
【성명의 영문표기】	LEE, Yoon Woo
【주민등록번호】	650525-1177713

**【우편번호】** 442-470  
**【주소】** 경기도 수원시 팔달구 영통동 현대아파트 725동 503호  
**【국적】** KR  
**【발명자】**  
**【성명의 국문표기】** 김현정  
**【성명의 영문표기】** KIM,Hyun Jung  
**【주민등록번호】** 731110-2011738  
**【우편번호】** 441-390  
**【주소】** 경기도 수원시 권선구 권선동 삼천리2차아파트 102-1108  
**【국적】** KR  
**【취지】** 특허법 제42조의 규정에 의하여 위와 같이 출원합니다. 대리인  
 이영필 (인) 대리인  
 이해영 (인)  
**【수수료】**  
**【기본출원료】** 20 면 29,000 원  
**【가산출원료】** 3 면 3,000 원  
**【우선권주장료】** 0 건 0 원  
**【심사청구료】** 0 항 0 원  
**【합계】** 32,000 원  
**【첨부서류】** 1. 요약서·명세서(도면)\_1통

**【요약서】****【요약】**

본 발명은 저밀도 패리티 검사를 이용한 부호화에 있어서 버스트 에러 발생시 높은 에러 정정율을 달성하기 위한 인터리빙 방법에 관한 것이다. 본 발명은, 밀도 패리티 검사 부호화에 사용되는 인터리빙 방법에 있어서, 패리티 검사 행렬에 기초하여 패리티 정보를 생성함으로써 하나 이상의 코드 워드 벡터를 생성하는 단계; 상기 생성된 코드 워드 벡터를, 상기 패리티 검사 행렬의 열 내에 존재하는 구성 성분 1 의 간격에 기초하여 결정된 크기를 가지는, 인터리빙 단위로 분할하는 단계; 및 상기 인터리빙 단위로 상기 하나 이상의 코드 워드 벡터를 인터리빙 하는 단계를 포함한다. 본 발명에 의하면, LDPC 부호화를 사용함에 있어 최적의 인터리빙 단위의 크기를 결정함으로써 에러 정정 신뢰도가 증가된다.

**【대표도】**

도 4

**【명세서】****【발명의 명칭】**

저밀도 패리티 검사 부호화를 위한 인터리빙 방법{Method of interleaving for low density parity check encoding}

**【도면의 간단한 설명】**

도 1은 통신 및 저장 매체 시스템에서 부호화 및 복호화의 개요도.

도 2는 저밀도 패리티 검사 부호화에서 패리티 검사 행렬과 생성된 코드 워드 벡터의 관계를 나타내는 도면.

도 3은 LDPC 부호화된 코드 워드 벡터내의 1의 위치와 버스트 에러의 크기와의 관계를 나타낸 도면.

도 4는 서로 다른 인터리빙 단위 크기를 가지는 코드 워드 벡터를 나타내는 도면.

도 5는 에러 정정 한계가 1비트인 경우 인터리빙 단위의 크기를 결정하는 일반적인 방법을 설명하는 도면.

도 6은 본 발명에 따른 인터리빙 단위를 적용하여 인터리빙 및 디인터리빙을 수행하였을 때 코드 워드 벡터의 변화를 나타내는 도면.

도 7은 에러 정정 한계가 1비트인 경우 에러 정정 신뢰율과 인터리브 단위 크기의 관계를 나타내는 도면이다.

## 【발명의 상세한 설명】

## 【발명의 목적】

## 【발명이 속하는 기술분야 및 그 분야의 종래기술】

- > 본 발명은 인터리빙 방법에 관한 것으로서, 보다 상세하게는 저밀도 패리티 검사를 이용한 부호화에 있어서 버스트 에러 발생시 높은 에러 정정율을 달성하기 위한 인터리빙 방법에 관한 것이다.
- > 무선통신분야 또는 광 기록재생 분야 등에서 사용되는 에러 정정 부호화 및 복호화 기술의 한 가지로서 저밀도 패리티 검사(Low Density Parity Check, 이하 LDPC 라 한다) 부호화 및 복호화 방법이 있다. LDPC 부호화는 1962년 갤러거(Gallager)에 의해 처음 제안되었으나, 당시의 기술력으로는 복호기의 구현이 어려워 잊혀져 오다 최근 맥케이(Mackay) 등에 의해 재발견되었다.
- > LDPC 부호화는 패리티 검사 행렬을 이용하여 패리티 정보를 생성하는 과정을 포함한다. 이 때 패리티 검사 행렬의 구성 성분 대부분은 0 이고 극히 일부 구성 성분만이 1 이다. 이러한 LDPC 부호화는 합곱 알고리즘을 사용하여 반복 부호를 수행함으로써 우수한 오류 성능을 가진다. 예를 들어, 부호어의 길이가  $10^6$  이며 부호율이  $1/2$  인 비정규 LDPC 부호는 터보 부호보다 섀넌(Shannon) 한계에 더욱 근접하는 성능을 가진다.
- > LDPC 부호화는 규칙적 LDPC 부호화(regular LDPC)와 불규칙적 LDPC 부호화(irregular LDPC)로 나뉜다. 규칙적 LDPC 는 부호화 및 복호화에 사용되는 패리티 검사 행렬이 포함하는 구성 성분 1 의 개수가 행(row) 및 열(column)마다 동일한 경우이고, 불규칙적 LDPC 는

그렇지 않은 경우이다. 규칙적 LDPC 부호화에서 각 행 및 열에 존재하는 "1"의 개수를 행 웨이트 및 열 웨이트라고 한다.

LDPC 부호화는 이하 수학적 식 1로 표현될 수 있다.

· 【수학적 식 1】  $H \cdot C_e = 0$

> H는 패리티 검사 행렬이고 0는 영행렬,  $\cdot$ 는 XOR 연산 및 모듈러 2 연산을 의미한다.  $C_e$ 는 코드 워드 벡터로서, 부호화 대상이 되는 코드 워드를 나타내는 열행렬(column matrix)이다. 코드 워드는 x비트의 메시지 워드  $x_1, x_2, \dots, x_x$ 와 p비트의 패리티 정보  $p_1, p_2, \dots, p_p$ 로 이루어진다.

> 패리티 정보  $p_1, p_2, \dots, p_p$ 는 각 메시지 워드  $x_1, x_2, \dots, x_x$ 가 수학적 식 1을 만족하도록 생성된다. 즉 패리티 검사 행렬 H와 행렬  $C_e$ 의 구성 성분중 부호화 대상인 메시지 워드의 이진값은 결정되어 있으므로 상기 수학적 식 1에 의해 패리티 정보  $p_i (i=1, 2 \dots p)$ 를 결정할 수 있다.

> LDPC 부호화에 대한 보다 상세한 설명은 "Good error correction codes based on very sparse matrices"(D.J.MacKay, IEEE Trans. on Information Theory, vol. 45, no.2, pp.399-431, 1999)"에 기술되어 있다.

> 인터리빙은 버스트 에러에 대처하기 위한 기술이다. 통신 또는 저장 매체 시스템에서, 신호가 채널을 통과할 때는 전송되는 신호의 특정 부분에만 편중되어 발생하는 버스트 에러가 발생할 수 있다. 이러한 버스트 에러는 통신 시스템에서는 전달 매체의 외부적인 원인에 의해, 저장 매체 시스템에서는 저장 매체의 스크래치등에 의해 발생한다. 이러한 버스트 에러는 전송되는 비트열의 특정 위치에 발생하기 때문에, 이러한 특정 위치에 존재하는 정보를 다른 위치에 분산시켜 놓았다가 수신단에서 복호화시 원래의 위치로 재위치시킨다면 에러가 발생한 위치에



의 에러 크기를 감소시킬 수 있다. 감소된 크기의 에러는 에러가 발생하지 않는 영역의 정보 예를 들면 패리티 정보 등을 이용하여 복원가능하다.

인터리빙 기술은 LDPC 부호화에도 적용가능하다. LDPC 에 적용되는 인터리빙 기술 중 하나는, 패리티 검사 행렬에 의해 생성된 코드 워드 벡터를 여러개 사용하여 에러 정정 블록을 생성한 후, 이 에러 정정 블록을 일정한 크기로 나누어서 인터리빙 하는 것이다. 그러나 종래의 인터리빙 방법을 그대로 적용함에 있어서, 좀 더 효율적인 인터리빙을 위한 인터리빙 단위의 크기에 대해서는 아무런 정보가 존재하지 않는다. 즉 LDPC 부호화에 종래의 인터리빙 방법을 적용함에 있어서, 어떠한 크기로 인터리빙을 하는 것이 LDPC 부호화의 버스트 에러 정정에 있어 효율적인가에 대해서는 알려진 것이 없다.

#### 【발명이 이루고자 하는 기술적 과제】

- > 따라서 본 발명은 전술한 문제점을 해결하기 위해 안출된 것으로서, LDPC 부호화를 사용함에 있어 최적의 인터리빙 단위의 크기를 결정함으로써 에러 정정 신뢰도를 증가시킨 인터리빙 방법을 제공하고자 한다.

#### 【발명의 구성 및 작용】

- > 전술한 과제를 해결하기 위한 본 발명은, 밀도 패리티 검사 부호화에 사용되는 인터리빙 방법에 있어서, 패리티 검사 행렬에 기초하여 패리티 정보를 생성함으로써 하나이상의 코드 워드 벡터를 생성하는 단계; 상기 생성된 코드 워드 벡터를, 상기 패리티 검사 행렬의 열 내에 존재하는 구성 성분 1 의 간격에 기초하여 결정된 크기를 가지는, 인터리빙 단위로 분할하는 단계; 및 상기 인터리빙 단위로 상기 하나이상의 코드 워드 벡터를 인터리빙 하는 단계를 포함하는 것을 특징으로 한다.

또한 본 발명은, 상기 생성된 코드 워드 벡터를 인터리빙 단위로 분할하는 단계는, 상기 패리티 검사 행렬의 열 내에 포함된 모든 구성 요소 1에 대해, 상기 구성 요소 1을 포함하나 다른 구성 요소 1은 포함하지 않는 최대 범위의 비트 길이를 추출하는 단계; 및 상기 추출된 비트 길이에 기초하여 상기 인터리빙 단위의 크기를 결정하는 단계를 포함하는 것을 특징으로 한다.

또한 본 발명은, 저밀도 패리티 검사 부호화에서 인터리빙 단위의 크기를 결정하는 방법으로서, 패리티 검사 행렬의 열 내에서 구성 성분이 1인 위치에 대응되는 코드 워드 벡터내의 코드 워드 비트를 유효 코드 워드 비트로 추출하는 단계; 상기 코드 워드 벡터내에서 상기 유효 코드 워드 비트 사이의 간격을 추출하는 단계; 상기 유효 코드 워드 비트사이의 간격에 기초하여 상기 인터리빙 단위의 크기를 결정하는 단계를 포함하는 것을 특징으로 한다.

- > 이하, 첨부된 도면을 참조하여 본 발명에 따른 바람직한 일 실시예를 상세히 설명한다.
- > 도 1은 통신 및 저장 매체 시스템에서 부호화 및 복호화의 개요도이다.
- > LDPC 인코더(110) 전송되어질 원래의 메시지 워드(111)를 수신하여 LDPC 인코딩함으로써 수 개의 코드 워드 벡터(121)를 생성한다. 각 코드 워드 벡터(121)는 메시지 워드(111)와 전술한 수학적 식 1을 만족하도록 생성된 패리티 정보를 포함한다. 인터리버(120)는 수 개의 코드 워드 벡터(121)를 수신하여 에러 정정 블록을 형성하고 이를 적절한 크기로 분할하고 적절한 위치에 분산시킴으로써 인터리빙된 비트열(131)을 생성한다. 통신 시스템에서, 이러한 인터리빙된 비트열(131)은 공기등의 전송 매체를 통해 전송될 것이고, 저장 매체 시스템에서, 이러한 인터리빙된 비트열(131)은 저장 매체에 기록된 형태로 재생기로 전달될 것이다.

수신단에서 또는 재생기에서, 디인터리버(130)는 인터리빙된 비트열(131)을 수신하여 이를 디인터리빙함으로써 원래의 코드 워드 벡터(141)를 생성한다. LDPC 디코더(130)는 코드 워드 벡터(141)를 수신하여 LDPC 복호화 알고리즘을 이용하여 원래의 메시지 워드(111)를 생성한다.

도 2는 저밀도 패리티 검사 부호화에서 패리티 검사 행렬과 생성된 코드 워드 벡터의 관계를 나타내는 도면이다.

LDPC 부호화는 패리티 검사 행렬  $H$  와 코드 워드 벡터  $C$  의 XOR 및 모듈러 연산의 결과가 영행렬이 되도록 코드 워드 벡터  $C$  내의 패리티 정보를 생성하는 과정이다. 하나의 코드 워드 벡터 내에 존재하는 패리티 정보  $p_1, p_2, \dots$  값을 알기 위해, 패리티 검사 행렬의 열의 수와 동일한 수의 방정식이 생성된다. LDPC 부호화는 이 방정식들을 풀어서 패리티 정보  $P_1, P_2, \dots$ 를 구하는 과정이다. 도 2에 나타난 예에서 패리티 검사 행렬의 열의 수가 10이므로, 10개의 방정식이 생성된다.

전술한 방정식들은 패리티 검사 행렬의 열과 코드 워드 벡터의 XOR 연산 및 모듈러 연산에 의해 생성되므로, 패리티 검사 행렬의 열내에서 구성 성분이 "1" 만이 방정식의 생성에 영향을 미친다. 따라서 패리티 검사 행렬의 하나의 열  $R_1, R_2, \dots$  각각이 가지는 "1" 만이 부호화에 영향을 미치고, 패리티 검사 행렬의 열이 가지는 "0" 은 부호화의 결과에 영향을 미치지 못한다.

> 도 2에서 패리티 검사 행렬의 열  $R_1$  의 빗금친 구성 성분(201,202,203)은 구성 성분이 1인 경우만을 나타낸 것이고, 코드 워드 벡터  $A$  의 빗금친 구성 성분(211,212,213)은 패리티 검사 행렬의 구성 성분(201,202,203)과 XOR 및 모듈러 연산되는 구성 성분을 나타낸 것이다. 이

들 달리 말하면, 코드 워드 벡터내에서 구성 성분(211,212,213)을 제외한 다른 구성 성분(도 2의 코드 워드 벡터 A에서 빗금치지 않은 구성 성분들)은 LDPC 부호화에 영향을 미치지 않는다.

LDPC 복호화 알고리즘은 수신된 코드 워드 벡터 A'로부터 원래의 코드 워드 벡터 A를 생성하는 과정이다. 현재 사용되는 모든 LDPC 복호화 알고리즘은 부호화에서 사용된 수학적 1을 사용한다. 즉 패리티 검사 행렬내의 열에 존재하는 1의 위치에 기초하여 복호화 과정을 수행한다. 이는 복호화 과정에서 패리티 검사 행렬의 구성 성분이 "1"인 위치에 대응하는 코드 워드 벡터의 코드 워드 비트(211,212,213)가 동일한 복호화 알고리즘으로 복호화됨을 의미한다.

- > 인터리빙 과정은 모든 코드 워드 벡터 A,B,C,...를 일정한 크기의 인터리빙 단위로 분할한 후 인터리빙 단위들을 미리결정된 규칙으로 서로 다른 위치에 위치시키는 과정이다. 인터리빙 과정에서 이러한 인터리빙 단위를 크기를 결정하는 것은 버스트 에러 발생시 에러 정정에 대한 신뢰도에 많은 영향을 미친다.
- > 만약 인터리빙 단위를 가장 크게 하여, 즉 인터리빙 과정 없이 코드 워드 벡터를 순차적으로 하여, 전송하거나 순서대로 저장 매체에 기록하면, 버스트 에러 발생시 에러가 발생한 위치에는 모두 동일한 코드 워드 벡터에 속하는 코드 워드 비트가 위치하기 때문에, 복호화시 특정 코드 워드 벡터 전체를 복호화 할 수 없는 문제가 생긴다. 또한 인터리빙 단위의 크기를 너무 작게하는 것은 디인터리빙시 복잡화 문제를 야기시키고, 에러 정정 블록의 크기로 인한 제한 때문에 그 구현이 용이하지 않다. 따라서 최적의 인터리빙 단위의 크기를 결정하는 것은 높은 에러 정정 신뢰도를 달성하는데 중요하다.
- ↳ 도 3은 LDPC 부호화된 코드 워드 벡터내의 1의 위치와 버스트 에러의 크기와의 관계를 나타낸 도면이다.

도 3에서, 빗금친 코드 워드 비트는 도 2에서 패리티 검사 행렬내의 "1"에 대응하는 위치에 있는 코드 워드 벡터내의 코드 워드 비트(211,212,213)로서, 이하 '유효 코드 워드 비트'라 정의한다. 만약 버스트 에러 E1이 발생하면, 버스트 에러에 의해 왜곡되는 코드 워드 비트는 3번째 비트부터 7번째 비트까지이다. 버스트 에러 E1에 의해 왜곡되는 코드 워드 비트는 하나의 유효 코드 워드 비트를 포함한다. 만약 버스트 에러 E2이 발생하면, 버스트 에러에 의해 왜곡되는 코드 워드 비트는 2번째 비트부터 8번째 비트까지이다. 즉 버스트 에러 E2에 의해 왜곡되는 코드 워드 비트는 두개의 유효 코드 워드 비트를 포함한다.

- > 인터리빙 단위의 크기는 코드 워드 벡터내에서 버스트 에러에 대해 영향을 받는 유효 코드 워드 비트의 개수와 관련된다. 이는 도 4 및 도 5를 참조하여 자세히 설명된다.
- > 도 4는 서로 다른 인터리빙 단위 크기를 가지는 코드 워드 벡터를 나타내는 도면이다. 도 4는 인터리빙 단위 크기와 코드 벡터내의 유효 코드 워드 비트의 관계를 나타낸다.
- > 도 4의 첫 번째 그림은 인터리빙 단위를 5비트로 선택한 경우이고, 두 번째 그림은 인터리빙 단위를 7비트로 선택한 경우이다. 아직 인터리빙은 수행하지 않은 상태이다. 인터리빙 단위 BI1, BI2, ... 은 저장 매체에 기록시 또는 채널을 통해 전송시 서로 다른 위치에 위치될 것이기 때문에(즉 인터리빙될 것이기 때문에) 동일한 버스트 에러의 영향을 받지 않는다. 따라서 인터리빙 단위보다 큰 크기의 버스트 에러가 발생하더라도 최대 인터리빙 단위 크기에만 에러가 영향을 미친다. 또한 본 실시예에서 에러 정정 한계는 1비트라고 가정한다.
- > 첫 번째 경우에서, 인터리빙 단위 BI1은 하나의 유효 코드 워드 비트를 포함한다. 인터리빙 단위 BI1이 위치한 지역에 버스트 에러가 발생하였고, 발생한 버스트 에러의 크기가 8비트라고 가정하자. 버스트 에러의 크기가 8비트이지만 인터리빙된 상태에서 버스트 에러의 영

향을 받을 것이므로, 디인터리빙 후의 인터리빙 단위 BI2 에는 BI1에 영향을 미친 버스트 에러가 영향을 미칠 수 없다.

- 도 3에서 설명한대로 LDPC 부호화에서 하나의 코드 워드 벡터내에서 서로 패리티 정보에 영향을 미치는 것은 유효 코드 워드 비트 들만이고, 인터리빙 단위 BI1 가 포함하는 유효 코드 워드 비트의 개수는 1이기 때문에, 코드 워드 벡터 A 에 대해 발생한 에러는 1비트 에러이다. 인터리빙 단위 BI2 즉 디인터리빙 후에 코드 워드 벡터 A 가 포함하는 에러는 1비트이다. 이 비트는 에러 정정 가능하다.
- > 두 번째 경우에서, 인터리빙 단위 BI'1 은 두개의 유효 코드 워드 비트를 포함한다. 첫 번째 경우와 동일하게, 인터리빙 단위 BI'1 이 위치한 지역에 버스트 에러가 발생하였고, 발생한 버스트 에러의 크기가 도 4에 나타난 바와 같이 8비트라고 가정하자. 첫 번째 경우와 마찬가지로 유효 코드 워드 비트의 개수가 2이므로 코드 워드 벡터 A 에 대해 발생한 에러는 2비트 에러이다. 즉 디인터리빙 후에 코드 워드 벡터 A 가 포함하는 에러는 2비트이다. 이 비트는 에러 정정 불가능하다.
- > 도 4에서, 화살표는 위 두 경우의 인터리빙 단위에 대해 동일한 버스트 에러의 영향을 받은 영역을 표시한 것이다. 두 경우 모두 8비트의 버스트 에러가 발생하였지만 첫 번째 경우에는 인터리빙 단위를 5비트로 하였으므로 5비트에만 영향을 받았고 두 번째 경우에는 인터리빙 단위를 7비트로 하였으므로 7비트에 영향을 받았다. 첫 번째 경우에 있어서 버스트 에러의 영향을 받은 나머지 3비트 및 두 번째 경우에 있어서 버스트 에러의 영향을 받은 나머지 1비트는 코드 워드 벡터 B, C.... 중 어느 하나의 코드 워드 벡터내에 위치할 것이다. 이러한 코드 워드 벡터 B, C ,... 중 어느 하나에 위치한 버스트 에러의 영향을 받은 코드 워드 비트는 서로 상이한 코드 워드 벡터에 속하기 때문에 코드 워드 벡터의 에러 정정과 무관하다.

- 요약하면, 인터리빙 단위의 크기는 각 인터리빙 단위가 에러 정정 한계 이내의 유효 코드 워드 비트를 가지도록 결정되어야 한다. 도 4의 예에서 에러 정정 한계는 1비트로 가정하였으므로 각 인터리빙 단위는 1개 이하의 유효 코드 워드 비트를 가져야 한다.
- 도 5는 에러 정정 한계가 1비트인 경우 인터리빙 단위의 크기를 결정하는 일반적인 방법을 설명하는 도면이다.
- 도 5에서, 아래에 수평으로 배열된 원형 도트(dot)들은 특정 코드 워드 벡터내의 코드 워드 비트를 의미한다. 이러한 원형 도트 빗금친 원형 도트는 유효 코드 워드 비트에 해당한다.  $n$  은 하나의 코드 워드 벡터의 길이,  $M$  은 유효 코드 워드 사이의 평균 거리,  $L$  은 하나의 유효 코드 워드 비트만을 포함하는 최대 비트수를 나타낸다.
- 전술한 바와 같이, 버스트 에러에 대한 에러 정정이 가능하기 위한 조건은, 각 인터리빙 단위가 에러 정정 한계 이내의 유효 코드 워드 비트를 가지도록 인터리빙 단위가 결정되어야 한다는 것이다. 결국 도 5에서 허용되는 인터리빙 단위 크기의 최대값은  $L$  이다.  $L$  값은 코드 워드 벡터마다 상이하고 또한 하나의 코드 워드 벡터내에서도  $L$  값의 범위가 어떠한 유효 코드 워드 비트를 포함하는지에 따라 상이하다.
- 그러나, 각 유효 코드 워드 비트사이의 평균 거리  $M$ 을 측정할 수 있다면, 오직 하나의 유효 코드 워드 비트만을 포함하는 최대 거리  $L$  값은  $M$  값의 두 배가 될 것이다. 결국 인터리빙 단위의 최대 크기  $B_{\text{Imax}}$  는  $M$  값과 이하 수학식 2 의 관계를 가진다.
- 【수학식 2】  $B_{\text{Imax}} = L \approx 2M$
- 여기서  $B_{\text{Imax}}$  는 인터리빙 단위의 최대 크기,  $L$  은 오직 하나의 유효 코드 워드 비트를 포함하는 최대 거리,  $M$  은 유효 코드 워드 비트간의 평균 거리이다.

- 여기서  $L$  값과  $2M$  값이 완전히 동일하지 않은 것은 유효 코드 워드 비트사이의 거리가 코드 워드 벡터에 따라 그리고 동일한 코드 워드 벡터내에서도 각 코드 워드 비트마다 상이하기 때문이다. 즉 패리티 검사 행렬의 "1" 사이의 거리는 패리티 검사 행렬의 각 열마다 그리고 동일한 열내에서도 상이하기 때문이다.
- > 특히 규칙적 LDPC 부호화에서 코드 워드 벡터내에 포함된 유효 코드 워드 비트의 수는 패리티 검사 행렬의 열 웨이트  $W_r$  와 동일하다. 또한 유효 코드 워드 비트간의 평균 거리는 코드 워드 벡터의 길이  $n$  를 코드 워드 벡터의 열 웨이트  $W_r$  로 나눈값과 동일하다. 따라서, 규칙적 LDPC 부호화에서 인터리빙 단위 크기의 최대값  $B_{\max}$  는 이하 수학적 3 의 관계를 가진다.
- > 【수학적 3】  $B_{\max} = L \simeq 2M = 2n/W_r$
- > 여기서  $n$  은 코드 워드 벡터의 길이이고,  $W_r$  은 패리티 검사 행렬의 열 웨이트이다.
- > 만약 에러 정정 한계가 1비트가 아니라 2 비트라면  $L = 3M$  이고, 3비트라면  $L = 4M, \dots$ ,  $k$  비트라면  $L = (k+1)M$  이 될 것이다. 결국 수학적 3 은 수학적 4 로 일반화된다.
- > 【수학적 4】  $B_{\max} = L = (k+1)M = (k+1)n/W_r$
- > 여기서  $k$  는 에러 정정 한계이고,  $n$  은 코드 워드 벡터의 길이,  $W_r$  은 열 웨이트이다.
- > 도 6은 본 발명에 따른 인터리빙 단위를 적용하여 인터리빙 및 디인터리빙을 수행하였을 때 코드 워드 벡터의 변화를 나타내는 도면이다.
- > 첫 번째 그림은 인터리빙 이전의 코드 워드 벡터  $A, B, C, \dots$ 를 나열한 비트열을 나타낸다. 인터리빙이전의 코드 워드 벡터  $A, B, C, \dots$  는 각각 인터리빙 단위  $A_1, A_2, A_3, \dots, B_1, B_2, B_3, \dots, C_1, C_2, C_3, \dots$ 를 포함한다.



두 번째 그림은 인터리빙 단위  $A_1, A_2, A_3, \dots, B_1, B_2, B_3, \dots, C_1, C_2, C_3, \dots$ 를 소정의 방식으로 인터리빙 한 후의 비트열을 나타낸다. 여기서 사용된 인터리빙 방식은 각 코드 워드 벡터의 인터리빙 단위들을 순차적으로 추출하여 인터리빙 하였다. 인터리빙 후의 비트열이 통신 채널을 통해 전송되거나 저장 매체에 저장된다.

세 번째 그림은 수신단 또는 재생기에서 디인터리빙한 후의 비트열을 나타낸다. 이 때 인터리빙 단위  $A_2, B_2, C_2$ 가 위치한 지역에 버스트 에러가 발생하였다고 가정하자. 버스트 에러는 인터리빙 단위  $A_2, B_2, C_2$ 를 인터리빙 단위  $EA_2, EB_2, EC_2$ 로 왜곡시켰다. 왜곡된 인터리빙 단위  $EA_2, EB_2, EC_2$ 는 디인터리빙 동작에 의해 원래 코드 워드 벡터내로 재워치된다.

네 번째 그림은 디인터리빙 후의 코드 워드 벡터 A의 내부 구성을 나타낸다. 이때 인터리빙 단위  $EA_2$ 는 하나의 유효 코드 워드 비트를 포함한다. 따라서 나머지 왜곡을 받지 않은 인터리빙 단위  $A_1, A_3, A_4, \dots$ 에 의해  $EA_2$ 는 에러 정정될 수 있다. 만약 인터리빙 단위 크기를 인터리빙 단위가 두 개이상의 유효코드 워드 비트를 포함하도록 결정하였다면, 인터리빙 단위  $EA_2$ 는 두 개이상의 유효 코드 워드 비트를 포함하기 때문에 디인터리빙 후의 코드 워드 벡터 A는 두 개 이상의 유효 코드 워드 비트를 포함하고 그 결과 1비트 에러 정정이 불가능할 것이다.

> 도 7은 에러 정정 한계가 1비트인 경우 에러 정정 신뢰율과 인터리빙 단위 크기의 관계를 나타내는 도면이다.

> 전술한 바와 같이, 본 발명의 사상에 의해 규칙적 LDPC 부호화에서 인터리빙 단위가 가질 수 있는 최대값은  $2n/W_r$ 이다. 따라서 가능한 인터리빙 단위 BI의 범위는  $1 < BI < 2n/W_r$ 이다. 인터리빙 단위의 최대값은  $n$ 값이 클수록 그리고  $W_r$ 값이 작을수록 증가한다. 도 7은  $2n/W_r$ 값을 기준으로 에러 정정 신뢰율이 급격히 떨어짐을 나타낸다.

$2n/W_r$  값보다 작은 값에서 에러 정정 신뢰율이 감소하는 것은 패리티 검사 행렬에서 "1"의 간격이 일정하지 않다는 점에 기인한다. 즉 유효 코드 워드 비트의 거리가 모든 코드 워드 벡터에서 또는 동일한 코드 워드 벡터내에서 일정하지 않기 때문에 우리가 평균값이라고 가정한  $n/W_r$  보다 작은 값을 가지는 유효 코드 워드 비트 거리가 존재할 수 있기 때문이다. 따라서 우리는  $2n/W_r$  보다 조금 작은 값을 인터리빙 단위 크기로 결정함으로써 에러 정정 신뢰도를 좀 더 증가시킬 수 있다. 이 때  $2n/W_r$  값과 최적 인터리빙 단위 크기  $BI_{opt}$ 의 차이  $D$ 는 패리티 검사 행렬에 포함된 "1" 분포의 균일도가 높을수록 그리고 "1"의 밀도가 낮을수록 즉 열 웨이트가 낮을수록 감소한다.

본 발명은 인터리빙 단위 크기에 대한 최적의 선택 방법을 제공하는 것이 목적이다. 이를 위해, 최선의 방법은 모든  $L$  값을 추출하여 이보다 작은 크기를 인터리빙 단위 크기로 결정하는 것이다. 그러나 근사화에 의해  $2M$  값을  $L$  값으로 간주하여 이보다 작은 크기를 인터리빙 단위 크기로 결정하는 방법이나, 보다 더 간단히  $2n/W_r$  보다 작은 값을 인터리빙 단위 크기로 결정하는 방법 역시 본 발명의 목적을 달성한다. 이러한 차선택들은 최선의 방법보다 에러 정정 신뢰율이 조금 감소할 뿐, 유효 코드 워드 비트 사이의 거리를 고려하지 않는 종래의 방법보다는 에러 정정 신뢰율이 향상되는 것은 분명하기 때문이다.

이제까지 본 발명에 대하여 그 바람직한 실시예들을 중심으로 살펴보았다. 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자는 본 발명이 본 발명의 본질적인 특성에서 벗어나지 않는 범위에서 변형된 형태로 구현될 수 있음을 이해할 수 있을 것이다. 그러므로 개시된 실시예들은 한정적인 관점이 아니라 설명적인 관점에서 고려되어야 한다. 본 발명의 범위는 전술한 설명이 아니라 특허청구범위에 나타나 있으며, 그와 동등한 범위 내에 있는 모든 차이점은 본 발명에 포함된 것으로 해석되어야 할 것이다.

【발명의 효과】

전술한 바와 같이 본 발명에 따르면, LDPC 부호화를 사용함에 있어 최적의 인터리빙 단위  
위의 크기를 결정함으로써 에러 정정 신뢰도를 증가시킨 인터리빙 방법이 제공된다.

**【특허청구범위】****【청구항 1】**

저밀도 패리티 검사 부호화에 사용되는 인터리빙 방법에 있어서,

패리티 검사 행렬에 기초하여 패리티 정보를 생성함으로써 하나이상의 코드 워드 벡터를 생성하는 단계;

상기 생성된 코드 워드 벡터를, 상기 패리티 검사 행렬의 열 내에 존재하는 구성 성분 1의 간격에 기초하여 결정된 크기를 가지는, 인터리빙 단위로 분할하는 단계; 및

상기 인터리빙 단위로 상기 하나이상의 코드 워드 벡터를 인터리빙 하는 단계를 포함하는 것을 특징으로 하는 인터리빙 방법.

**【청구항 2】**

제 1 항에 있어서, 상기 생성된 코드 워드 벡터를 인터리빙 단위로 분할하는 단계는,

상기 패리티 검사 행렬의 열 내에 포함된 모든 구성 요소 1에 대해, 상기 구성 요소 1을 포함하나 다른 구성 요소 1은 포함하지 않는 최대 범위의 비트 길이를 추출하는 단계; 및

상기 추출된 비트 길이에 기초하여 상기 인터리빙 단위의 크기를 결정하는 단계

를 포함하는 것을 특징으로 하는 인터리빙 방법.

**【청구항 3】**

제 2 항에 있어서, 상기 인터리빙 단위의 크기를 결정하는 단계는,

상기 추출된 비트 길이의 평균값을 상기 인터리빙 단위의 크기로 결정하는 단계

를 포함하는 것을 특징으로 하는 인터리빙 방법.

**【청구항 4】**

제 2 항에 있어서, 상기 인터리빙 단위의 크기를 결정하는 단계는,  
상기 추출된 비트 길이의 최소값을 상기 인터리빙 단위의 크기로 결정하는 단계  
를 포함하는 것을 특징으로 하는 인터리빙 방법.

**【청구항 5】**

제 1 항에 있어서, 상기 생성된 코드 워드 벡터를 인터리빙 단위로 분할하는 단계는,  
상기 패리티 검사 행렬의 열 내에 존재하는 모든 구성 성분 1 사이의 간격을 추출하는  
단계;  
상기 추출된 간격의 평균값을 계산하는 단계; 및  
상기 추출된 간격의 평균값의 두 배에 해당하는 비트 길이를 상기 인터리빙 단위의 크  
기로 결정하는 단계  
를 포함하는 것을 특징으로 하는 인터리빙 방법.

**【청구항 6】**

제 1 항에 있어서, 상기 생성된 코드 워드 벡터를 인터리빙 단위로 분할하는 단계는,  
상기 코드 워드 벡터의 길이와 상기 코드 워드 벡터의 열 웨이트에 기초하여 상기 인터  
리빙 단위의 크기를 결정하는 단계; 및  
상기 결정된 크기를 가지는 인터리빙 단위로 상기 코드 워드 벡터를 분할하는 단계  
를 포함하는 것을 특징으로 하는 인터리빙 방법.

**【청구항 7】**

제 6 항에 있어서, 상기 인터리빙 단위의 크기를 결정하는 단계는, 상기 코드 워드 벡터의 길이를 열 웨이트로 나눈 값의 두 배에 해당하는 값보다 적은 값을 인터리빙 단위의 크기로 결정하는 것을 특징으로 하는 인터리빙 방법.

**【청구항 8】**

저밀도 패리티 검사 부호화에서 인터리빙 단위의 크기를 결정하는 방법으로서,

패리티 검사 행렬의 열 내에서 구성 성분이 1 인 위치에 대응되는 코드 워드 벡터내의 코드 워드 비트를 유효 코드 워드 비트로 추출하는 단계;

상기 코드 워드 벡터내에서 상기 유효 코드 워드 비트 사이의 간격을 추출하는 단계;

상기 유효 코드 워드 비트사이의 간격에 기초하여 상기 인터리빙 단위의 크기를 결정하는 단계

를 포함하는 것을 특징으로 하는 크기 결정 방법.

**【청구항 9】**

제 8 항에 있어서, 상기 인터리빙 단위의 크기를 결정하는 단계는,

상기 유효 코드 워드 비트 사이의 간격의 최소값의 두 배보다 적은 값을 인터리빙 단위의 크기로 결정하는 단계를 포함하는 것을 특징으로 하는 크기 결정 방법.

**【청구항 10】**

제 8 항에 있어서, 상기 인터리빙 단위의 크기를 결정하는 단계는,

상기 유효 코드 워드 비트 사이의 간격의 평균값의 두 배보다 적은 값을 인터리빙 단위의 크기로 결정하는 단계를 포함하는 것을 특징으로 하는 크기 결정 방법.

## 【청구항 11】

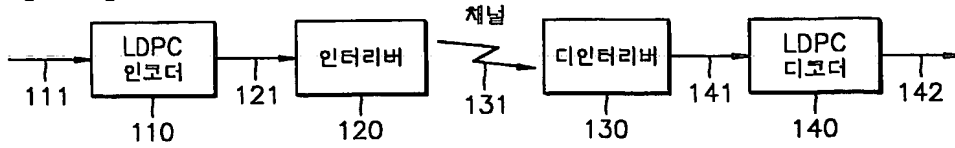
제 10 항에 있어서, 상기 유효 코드 워드 비트 사이의 간격의 평균값은 상기 코드 워드 벡터의 길이를 상기 코드 워드 벡터의 열 웨이트로 나눈 값인 것을 특징으로 하는 크기 결정 방법.

## 【청구항 12】

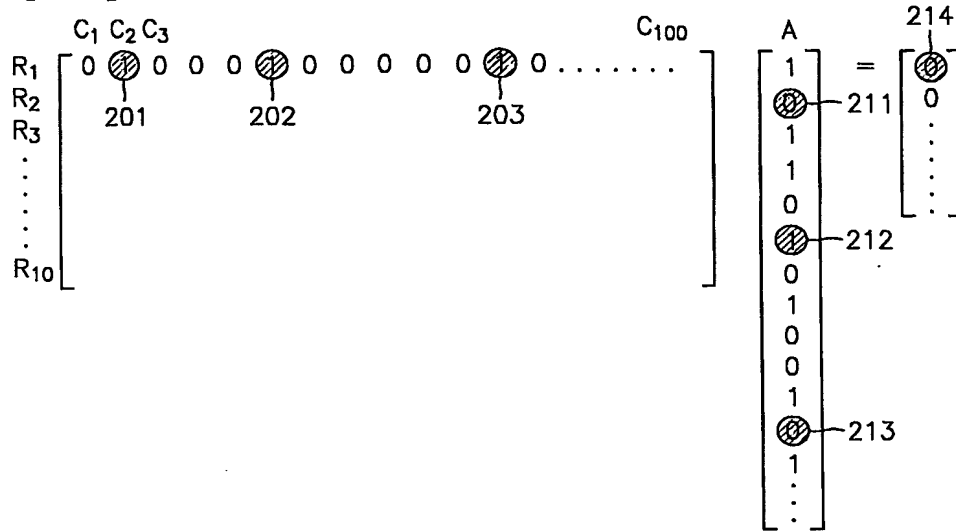
제 8 항에 있어서, 상기 인터리빙 단위의 크기를 결정하는 단계는 수식  $1 < BI < 2n/Wr$  을 만족하는 크기를 인터리빙 단위의 크기로 결정하고, 여기서 BI 는 인터리빙 단위의 크기, n 은 상기 코드 워드 벡터의 길이, Wr 은 상기 코드 워드 벡터의 열 웨이트인 것을 특징으로 하는 크기 결정 방법.

## 【도면】

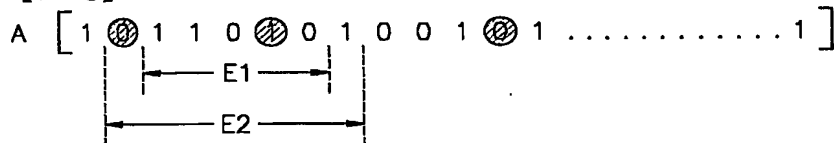
【도 1】



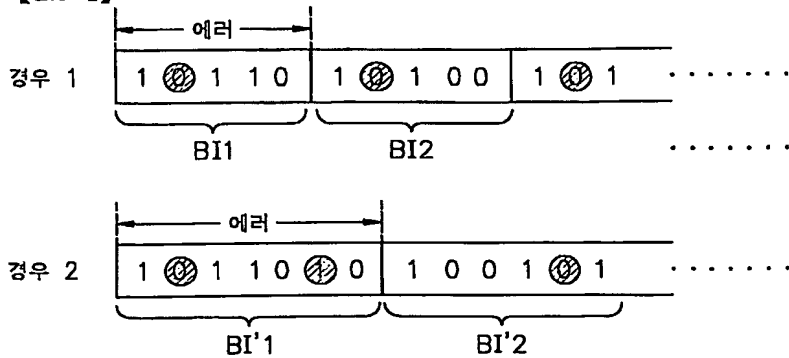
【도 2】



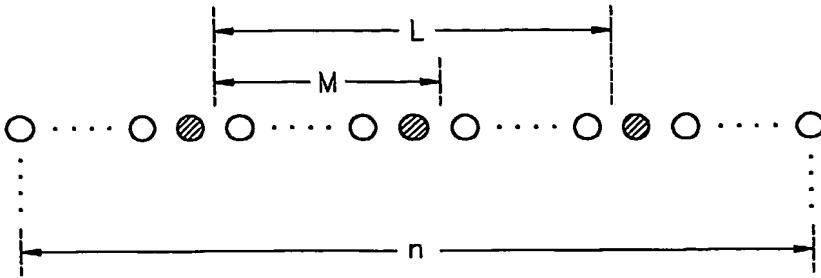
【도 3】



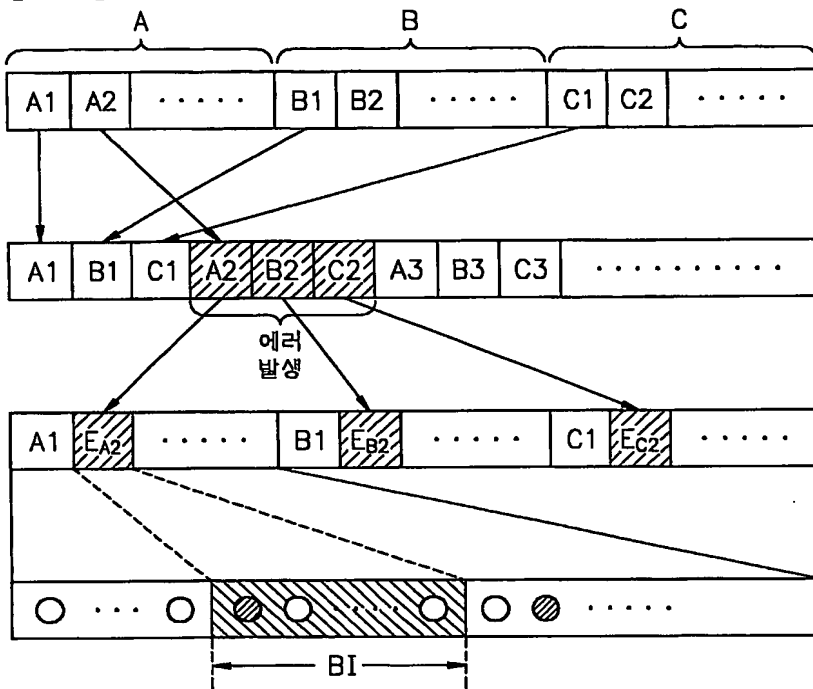
【도 4】



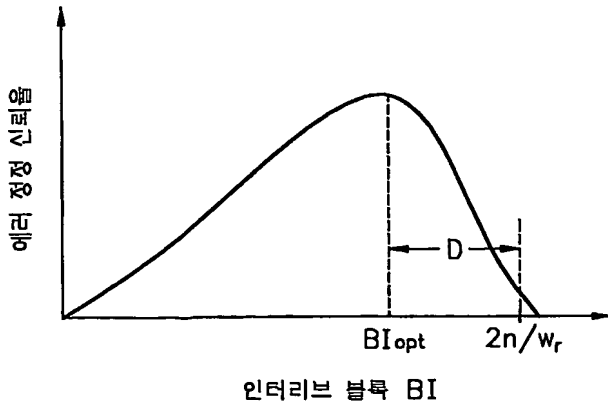
【도 5】



【도 6】



【도 7】



# Document made available under the Patent Cooperation Treaty (PCT)

International application number: PCT/KR04/003061

International filing date: 25 November 2004 (25.11.2004)

Document type: Certified copy of priority document

Document details: Country/Office: KR  
Number: 10-2003-0086048  
Filing date: 29 November 2003 (29.11.2003)

Date of receipt at the International Bureau: 03 January 2005 (03.01.2005)

Remark: Priority document submitted or transmitted to the International Bureau in compliance with Rule 17.1(a) or (b)



World Intellectual Property Organization (WIPO) - Geneva, Switzerland  
Organisation Mondiale de la Propriété Intellectuelle (OMPI) - Genève, Suisse